Indian J.Sci.Res. 17(2): 122-125, 2018

ISSN: 0976-2876 (Print)
ISSN: 2250-0138(Online)

TIME EFFICIENT METHODS FOR IMPROVING DUPLICATE DETECTION AND QUALITY ON LARGE DATASETS

[1]K.Santhosh
[1]Assistant Sales Executive, ICICI BANK, Two wheeler JSP HONDA Showroom, Hyderabad.

Abstract—One of the serious issues faced in many applications with personal details management, client affiliation management, data processing, etc is duplicate detection. This survey deals with the varied duplicate record detection techniques in each small and huge datasets. To discover the duplicity with less time of execution and also without disturbing the dataset quality, ways like Progressive blocking and Progressive Neighborhood are used. Progressive sorted neighborhood method also known as PSNM is used during this model for finding or detecting the duplicate in a parallel approach. Progressive blocking algorithm works on massive datasets wherever finding duplication needs vast time. These algorithms are used to enhance duplicate detection system. The efficiency can be doubled over the standard duplicate detection method using this algorithm. Several different ways of data analysis are studied here with varied approaches for duplicate detection.

keywords-- Data Duplicity Detection, PSNM, PB, Data Mining, Sorted Neighborhood Method, Progressive De-duplication.

I. Introduction

Data mining is also known as KDD or data discovery in database. The idea of data mining evolved from many researches that include statistics, database systems, machine learning concepts, neural networks, visualization, rough set, etc. each ancient and latest areas like businesses, sports, etc use the knowledge mining ideas. For translating the raw data into valuable information, the companies use a method. By knowing the small print concerning the purchasers and by developing efficient promoting policies, the sales and prices will be increased or decreased in the companies. The efficient collection of data, deposition a laptop process all has their influence on data processing ideas. The data is that the most essential necessary plus of any company however incase the data is modified or a unhealthy data entry is made certain errors like duplicate detection arises. Duplicate Detection Problems: Duplicate detection denotes to the method of recognizing different representations of the real world objectives present in associate information source. it is not possible to ignore many qualities of duplicate detection like effectiveness and scalability owing to the database size. There are two features in the problems of duplicate detection which are as follows: many representations usually are not same and have sure differences like misspelling, missing values, modified addresses, etc that makes the detection of duplicates very tough. The detection of duplicates is very expensive as a result of the comparison among all attainable duplicate pairs is needed. Progressive duplicate detection algorithms are as follows: PSNM or Progressive Sorted Neighborhood technique operating over clean and tiny datasets metal or Progressive blocking working over unclean and enormous datasets.

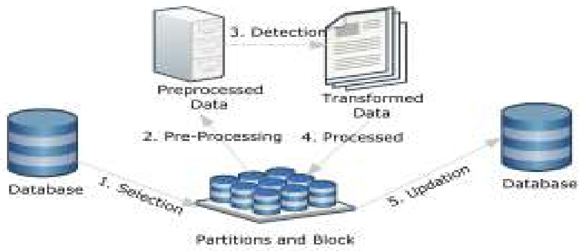


Figure 1: System Architecture

Above system architecture explains the method of duplicate data detection using progressive mechanism. This architecture is discussed in detail in section three of this paper. Definitions: Duplicate Detection: it is the process of recognizing many representations during a matched planet item. Data Cleaning: it is referred to as data scouring that denotes a process of detection, correction and removal of corrupted and inappropriate records present in the record sets, databases, tables, etc. Progressiveness: It progress the results, efficiencies and scalability of the algorithms used in this existing model. Techniques like window interval, look ahead, partition caching, and Magpie kind are used for delivering the results quicker. Entity Resolution: it is additionally known as de-duplication or record linkage that identifies the accounts equivalent to similar entity of a real-world. Pay-As-You-Go: It is a technique wherever the candidate pairs are theoretically ordered by the matching chances. Then comparison on records using the match pairs are performed using the ER algorithm.

II. Related Work

Much research on duplicate detection additionally referred to as entity resolution and by several alternative names, focuses on combine selection algorithms that strive to maximize recall on the one hand and efficiency on the

[1] Corresponding Author

other hand. The most distinguished algorithms during this area are blocking and also the organized neighborhood method (SNM) reconciling techniques. Previous publications on duplicate detection typically specialize in reducing the general runtime. Thereby, some of the projected algorithms are already capable of estimating the quality of comparison candidates. The algorithms use this information to decide on the comparison candidates additional carefully. For identical reason, different approaches utilize accommodative windowing techniques, which dynamically modify the window size depending on the quantity of recently found duplicates. These adaptive techniques dynamically improve the potency of duplicate detection, however in distinction to our progressive techniques, they need to run for certain periods of time and cannot maximize the efficiency for any given time slot Progressive procedures. In the most recent few years, the profitable need for progressive algorithms additionally initiated some concrete studies during this domain. As an example, pay-as you-go algorithms for info integration on giant scale datasets have been presented. Different works introduced progressive data cleansing algorithms for the analysis of sensor data streams. However, these approaches cannot be applied to duplicate detection. Xiao et al projected a top-k similarity be part of that uses a special index structure to estimate promising comparison candidates. This approach more and more resolves duplicates and additionally eases the parameterization problem. though the results of this approach is similar to our approaches (a list of duplicates almost ordered by similarity), the focus differs: Xiao et al. notice the top-k most similar duplicates regardless of however long this takes by weakening the similarity threshold; we discover as several duplicates as possible in a given time. That these duplicates are additionally the most similar ones is a facet effect of our approaches.

III. Frame Work

The projected solution uses two sorts of novel algorithms for progressive duplicate detection, that are as follows: PSNM – It is known as Progressive sorted neighborhood technique and it is performed over clean and small datasets. PB – it is called Progressive blocking and it is performed over dirty and huge datasets. Each these algorithms improve the efficiencies over vast datasets. The most aim of this paper is to observe the duplicate information within the different massive and small datasets as a parallel. During this paper, we tend to are detecting the duplicates on dataset. To observe duplicate data within the dataset, we tend to follow the three steps, pair selection, pair wise comparison, Clustering.

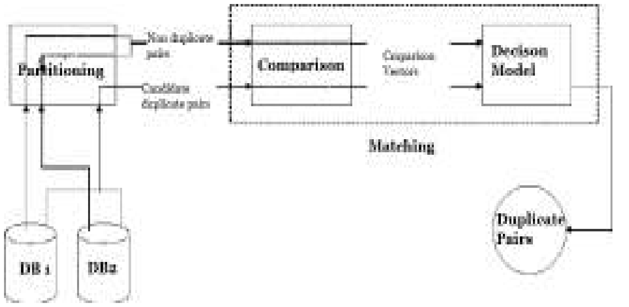


Figure2: Proposed System Architecture

In the above architecture diagram, we tend to take some datasets and therefore the start, we tend to are partition our complete dataset. Partition nothing however if we tend to provides a partition size=30 then this suggests, we tend to are keeping thirty records in each partition. When partition the dataset, we will perform the algorithm on the dataset. There in sorting, it will compare the duplicates as combine wise comparison. When comparison it will display the duplicate pairs to us. Dataset Overview: during this paper, we tend to are detecting the duplicates on CD dataset. It contains 9763 records and these records associated with the music and audio CDs. This dataset contain some attributes like, ID, artist, category, genre, cd-extra, and year. From these attributes we will get some attributes as sorting keys by using attribute concurrency methodology. If we tend to choose "artist" as a sorting key then the process done only supported the artist related data solely and when completion of processing it display the duplicate text of artist attribute from dataset.

Sorting Key:

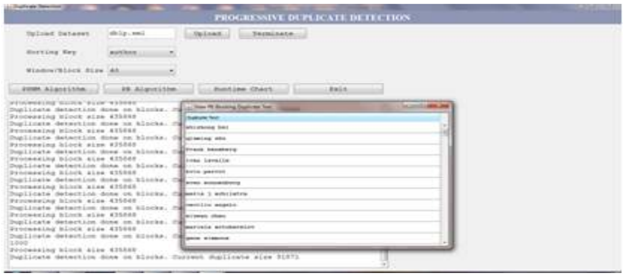
Importance of Sorting Key: Importance of this sorting secret is, generally giant dataset contains hundred thousand and thousands of records. For each time scan complete dataset and observe the all duplicates within the dataset is not potential. In typically user desires observe the duplicate data and observe the duplicate count on only specific data. During this kind of things, we want a sorting key without kindling key it is difficult to sort the information from dataset. To sorting the dataset, we tend to are using magpie sorting. During this sorting we tend to are choosing one sorting key. To pick out the most effective key for sorting we tend to are exploitation attribute concurrency methodology.

Sorting Key Selection: the most effective key for locating the duplicate is usually hard to spot. Choosing sensible keys can increase the progressiveness. Here all the records are taken and checked as a parallel processes thus on reduce average execution time. The records are kept in multiple resources once splitting. The intermediate duplication results are intimated instantly once found in any resources and came back to the most application. Thus the time utilization is reduced. Resource expenditure is same as offeredscheme but the data is kept in multiple resource memories.

Parallel Processing Method: data processing means that we tend to execute the quantity of processes at a time which means parallel this is often caused by exploitation some concurrency ways. In this technique initial we tend to are partition the dataset complete dataset. These concurrency ways execute the all partitions of the dataset at a time to reduce the execution time of the method. This projected methodology selects the sorting key from dataset by using attribute concurrency methodology. And it additionally takes the window/block size to partition the entire dataset. Basically, our projected system extended by ancient Progressive Sorted Neighborhood methodology (PSNM) and Progressive Block (PB) for that reason we want to allow the window size as partition size. Based on these sorting key and window size, the data processing technique executes the all partitions of the dataset and it additionally show the data processing time of the projected technique.

IV. Experimental Results

In our experiments, we are taking large dataset to detecting the duplicates. To detecting, first we have to select the sorting key. This key is selected by using attribute concurrency method. Through this method we can select the best key to sorting from uploaded dataset. This sorting key selection is common to either PSNM orProgressive Sorted Neighborhood Method and PB or Progressive Block algorithms. In PSNM or Progressive Sorted Neighborhood Method we are selecting window size and based that window size and sorting only it will detect the duplicates in the datasets.



Upload Dataset: dblp.xml
Sorting Key: authors

Figure 3

In Progressive Block or PB we are selecting block size as well as sorting key. These two algorithms works as parallel the duplicates are displayed in the milliseconds.

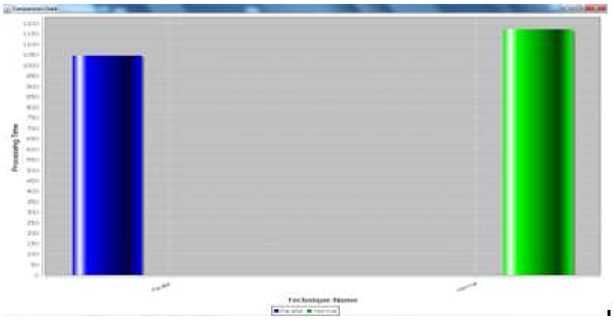


Figure 4

The above chart shows that the comparison between the parallel processing time and normal processing time. From our experiments we can prove that our proposed algorithms are time efficient and scalable approaches.

V. Conclusion

The progressive sorted neighborhood technique and progressive blocking algorithms increase the efficiency of duplicate detection for things with restricted execution time they dynamically modification the ranking of comparison candidates based on intermediate results to execute promising comparisons initial and less promising later. This paper surveys different analysis papers that projected numerous algorithms for detection of duplicate records. The progressive algorithms of duplicate detection are used to overcome disadvantages in varied analysis papers.

References

[1] "Data Mining Curriculum". ACM SIGKDD. 2006-04-30. Retrieved 2014-01-27.

[2] Fayyad, Usama; Piatetsky-Shapiro, Gregory; Smyth, Padhraic (1996). "From Data Mining to KnowledgeDiscovery in Databases" (PDF). Retrieved 17 December 2008.

[3] Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome (2009). "The Elements of Statistical Learning: Data Mining, Inference, and Prediction". Retrieved 2012-08-07.

[4] Witten, Ian H.; Frank, Elbe; Hall, Mark A. (30 January Data Mining: Practical Machine Learning Tools and Techniques (3 Ed.). Elsevier. ISBN 978-0-12-374856-0.

[5] Think Before You Dig: Privacy Implications of Data Mining & Aggregation, NASCIO Research Brief, September 2004

[6] Clifton, Christopher (2010). "Encyclopedia Britannica: Definition of Data Mining". Retrieved 2010-12-09.

[7] M. A. Hernández and S. J. Stolfo, "Real-world data is dirty: Data cleansing and the merge/purge problem," Data Mining and Knowledge Discovery, vol. 2, no. 1, 1998.

[8] Thorsten Papenbrock, ArvidHeise, and Felix Naumann,' Progressive Duplicate Detection'IEEE Transactions on Knowledge and Data Engineering(TKDE),vol . 25, no. 5, 2014.

[9] A.K. Elmagarmid, P. G. Ipeirotis, and V. S.Verykios, "Duplicate record detection: Asurvey," IEEETransactions on Knowledge and Data Engineering (TKDE), vol. 19, no. 1, 2007.S. E. Whang, D. Marmaros, and H. Garcia-Molina, "Pay-as-you-go entity resolution," IEEE Transactions on Knowledge and Data Engineering (TKDE), vol. 25, no. 5, 2012.

[10] "Adaptive windows duplicate-detection," in Proceedings of International Conference on Data Engineering (ICDE), 2012.

[11] U. Draisbach and F. Neumann, "A generalization ofblockingandwindowing algorithms for duplicate detection." In International Conference on Data and Knowledge Engineering (ICDKE), 2011.

[12] L. Kolb, A. Thor and E. Rahm, "Parallel sorted neighborhood blocking withmap-reduce," in Proceedings of the Conference Datenbanksysteme in Büro, Technik und Wissenschaft (BTW), 2011.

[13] C. Xiao, W. Wang, X. Lin and H. Shang, "Top-k set similarity joins," in Proc. IEEE Int. Conf. Data Eng., 2009, pp. 916–927.

[14] P. Indyk, "A small approximately min-wise independent family of hash functions," in Proc. 10th Annu. ACM-SIAM Symp. Discrete Algorithms, 1999, pp. 454–456.

[15] U. Draisbach and F. Naumann, "A generalization of blocking and windowing algorithms for duplicate detection," in Proc. Int. Conf. Data Knowl. Eng., 2011, pp. 18–24.